

PB-229 562

A NUMERICAL SOLUTION OF THE TWO-DIMENSIONAL
DIFFUSION EQUATION IN SHEAR FLOW

Jean F. Coudert

California Institute of Technology

Prepared for:

Federal Water Quality Administration

June 1970

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

A NUMERICAL SOLUTION OF THE TWO-DIMENSIONAL
DIFFUSION EQUATION IN A SHEAR FLOW

by

Jean F. Coudert

Project Supervisor:

Norman H. Brooks
Professor of Environmental Science
and Civil Engineering

Supported by Research Grant No. 16070 DGY
Federal Water Quality Administration
U. S. Department of the Interior

W. M. Keck Laboratory of Hydraulics and Water Resources
Division of Engineering and Applied Science
California Institute of Technology
Pasadena, California

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
U S Department of Commerce
Springfield VA 22151

Technical Memorandum 70-7

June 1970

TABLE OF CONTENTS

	<u>Page</u>
I Abstract	1
II Introduction	2
III Basic Equations	3
IV Description of the Numerical Method	5
V Initial Value	8
VI Velocity Profile	12
VII Diffusivity	13
VIII Program Check	15
IX Results for Different Cases	16
X Conclusion	17
Acknowledgment	19
Figures:	
1 Geometric layout	3
2 (a) (b) The numerical scheme	6
3 The numerical scheme	7
4 The geometry of the initial value	9
5 Series approximation to the δ -function	10
6 Side lobe magnitude for $P = 198; 199$	12
7 Concentration profiles	17
8 Results of JØB 113	20
9 (a) (b) (c) (d) (e) (f) Effect of changing initial input elevation	21-26

TABLE OF CONTENTS (Cont'd)

	<u>Page</u>
Tables:	
1 Solutions by previous workers	2
2 Ratios of M/m for N = 200.	11
3 Results of JØB 113	16
4 Variation of source height	17
References	27,28
Appendix A Computer Program	
Appendix B Sample Problem	

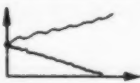
I. ABSTRACT

A numerical method developed in the last two years is used to solve the diffusion equation with a δ -function as initial value. The two-dimensional case of a steady line source in a stream is considered. The velocity profile is logarithmic and the diffusivity profile is parabolic but any profile could be used. The source can be set anywhere from the bed up to the free surface. The results are given in a dimensionless form and a significant set of plots is automatically produced. The method could be easily applied to more complicated cases (unsteady, point source, etc.).

II. INTRODUCTION

The problem of the diffusion of a substance from a line source in a river or in the atmosphere has already been studied in many different cases. However most of the solutions which have been given are analytical and they assume either uniform velocity and diffusivity profiles or power law profiles. For example, the following solutions have already been obtained (this is not exhaustive).

Table 1.
Solutions by Previous Workers

State	Doman	Velocity	Diffusivity	Reference
instantaneous line source		linear	constant	(1) P. G. Saffman
instantaneous line source		linear	constant	(1) P. G. Saffman
steady line source		power law	power law	(2) O. G. Sutton
		linear	constant	(6) A. Okubo

On the other hand, experiments have shown that the diffusion process is strongly related to the dispersion caused by the shear (Ref. 3). The influence of the variable diffusivity is much more difficult to investigate as no effective measurement can be made.

This discussion shows that the analytical tool does not provide help for such an investigation as the actual velocity profiles (logarithmic) and diffusivity profiles (parabolic) cannot be taken into account.

The goal of this study is to provide to the investigators a numerical method to solve such a problem. This method has been developed in the last two years by Dr. O. H. Kreis (Ref. 4) and some other people and solves any parabolic type equations provided the initial value can be expanded in a Fourier series (that is, all the generalized functions) (Ref. 5).

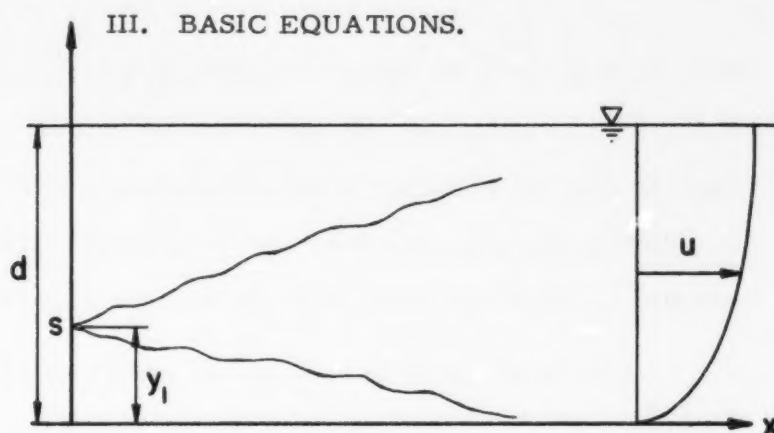


Fig. 1 Geometric layout.

We study the problem of the steady release of a substance into an open channel as shown on Fig. 1. The source is a line source and the problem is supposed to be two-dimensional.

The basic equation for the diffusion is (Ref. 10):

$$u(y) \frac{\partial C}{\partial x} = \frac{\partial}{\partial y} \left(\epsilon_y \frac{\partial C}{\partial y} \right) + \frac{\partial}{\partial x} \left(\epsilon_x \frac{\partial C}{\partial x} \right), \quad (1)$$

where ϵ_y and ϵ_x are diffusivities and are functions of y . C is the concentration, in weight per volume.

The boundary conditions are

$$\begin{aligned} \frac{\partial C}{\partial y}(x, 0) &= 0, \\ \frac{\partial C}{\partial y}(x, d) &= 0, \end{aligned}$$

$$C(y, 0) = \frac{S}{\bar{U}d} \delta(y-y_1) ,$$

S = amount (by weight) of substance released per unit time and unit width.

$\bar{U}$ = mean velocity,

δ = δ function,

The domain is defined by

$$0 \leq y \leq d,$$

$$0 \leq x.$$

Now, it is possible to neglect the term $\frac{\partial}{\partial x} \left(\epsilon_x \frac{\partial C}{\partial x} \right)$ as the diffusion in the x -direction is certainly less important for the spreading of the substance (Ref. 9) than the diffusion in the y -direction. However, there is no way to estimate the error involved and we will show that the complete solution would not be more difficult to solve than the simplified one.

$U(y)$ and $\epsilon_y(y)$ have to be known either experimentally or analytically.

In the problem studied, $U(y)$ is chosen as a logarithmic profile and $\epsilon_y(y)$ as a parabolic one.

In order to get results easier to read, Eq. (1) is written in the dimensionless form

$$\frac{U(y)}{U_*} \frac{\partial \left(\frac{C}{S/\bar{U}d} \right)}{\partial \left(\frac{x}{d} \right)} = \frac{\partial}{\partial \left(\frac{y}{d} \right)} \left(\frac{\epsilon_y(y)}{U_* d} \frac{\partial \left(\frac{C}{S/\bar{U}d} \right)}{\partial \left(\frac{y}{d} \right)} \right), \quad (2)$$

where

S = amount (by weight) of substance discharged per unit time ,

$\bar{U}$ = mean velocity,

U_* = shear velocity,

C = concentration in weight per volume.

In this notation, the initial value is

$$\frac{C(y)}{S/\overline{U_d}} = \delta(y-y_1) .$$

IV. DESCRIPTION OF THE NUMERICAL METHOD

There are two difficulties in solving numerically the problem described in the Introduction:

- the coefficients vary substantially close to the boundaries. This can introduce an instability in the difference scheme if the scheme is not designed for such cases.
- the initial value is not a smooth and bounded function (the function $\delta(y)$ and its first derivative are both generalized functions).

The classical way to work out the problem is to assume that the coefficients are constant over some domain close enough to the source and to compute the analytical solution. Then far enough from the source, a smooth solution can be computed and used as initial value for the numerical scheme with variable coefficients.

The trouble in such a method is that the result depends on where the matching of the analytical method and the numerical one takes place. To get a smooth enough initial value, this matching has to take place far enough from the source. Therefore, the possible effect of the variable coefficients near the source is omitted.

There has recently been developed a special kind of numerical method which can solve such problems, even very close to the source. The method is still unpublished (Ref. 4).

The Crank Nicholson scheme is used inside the domain and the initial function is replaced by its Fourier series truncated at the right frequency (this condition will be explained later on). The initial truncated Fourier series has to fulfill the boundary conditions, otherwise a singular point is introduced in the domain.

Two grids are defined on the domain as shown in Figs. 2a and 2b.

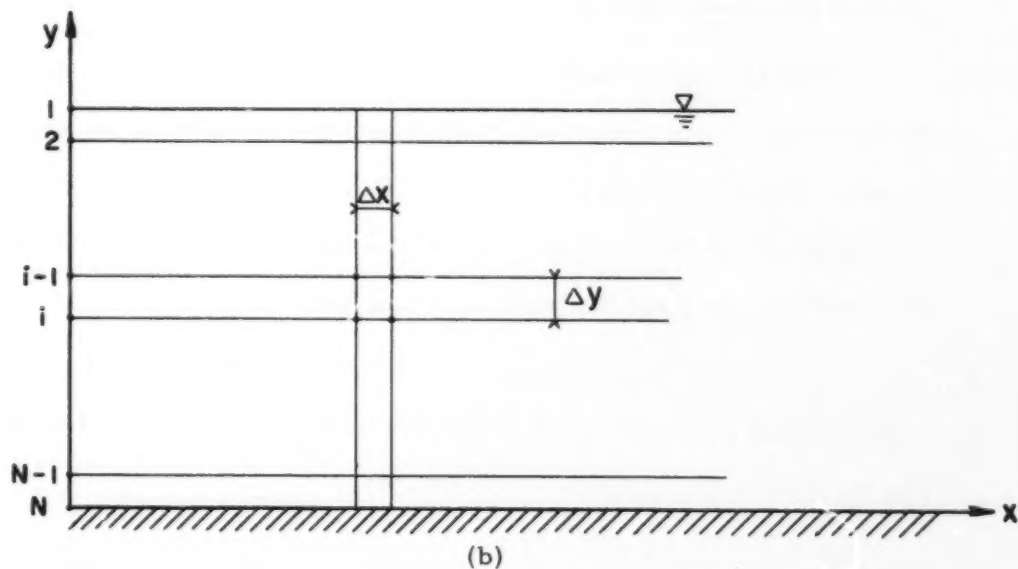
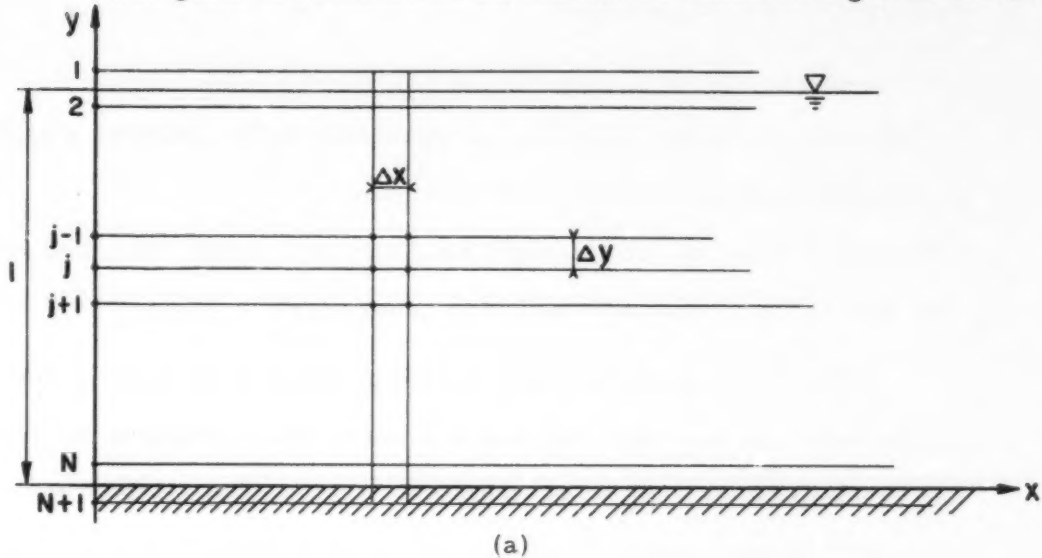


Fig. 2 The numerical scheme.

Let us set $C_j = C(y_j)$.

The Crank Nicholson (Ref. 11) scheme is

$$U_j \frac{C_j(x+\Delta x) - C_j(x)}{\Delta x} = \frac{1}{2} \left\{ \frac{\epsilon_{i-1} \frac{C_{j-1}(x+\Delta x) - C_j(x+\Delta x)}{\Delta y} - \epsilon_i \frac{C_j(x+\Delta x) - C_{j+1}(x+\Delta x)}{\Delta y}}{\Delta y} + \frac{\epsilon_{i-1} \frac{C_{j-1}(x) - C_j(x)}{\Delta y} - \epsilon_i \frac{C_j(x) - C_{j+1}(x)}{\Delta y}}{\Delta y} \right\}, \quad \begin{matrix} i = j \\ j = 2, 3, \dots, N \end{matrix} \quad (3)$$

The boundary conditions are

$$C_1 = C_2; \quad C_N = C_{N+1}.$$

The scheme is illustrated in Fig. 3.

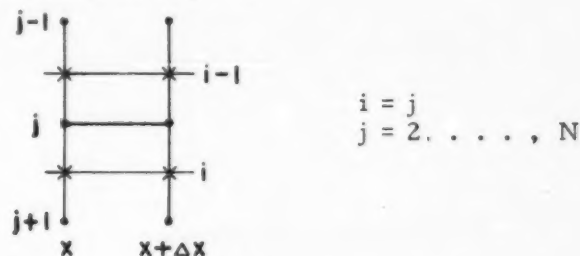


Fig. 3 The numerical scheme.

We can use a matrix-vector notation.

We set

$$C(x) = \begin{bmatrix} C_1(x) \\ C_2(x) \\ \cdot \\ \cdot \\ C_N(x) \\ C_{N+1}(x) \end{bmatrix}.$$

The iteration to find $C(x+\Delta x)$ from $C(x)$ can be written as

$$C(x+\Delta x) = \begin{bmatrix} 1 & -1 & & & \\ a_{12} & a_{22} & a_{32} & & 0 \\ & & & & \\ & & & & \\ 0 & & a_{1N} & a_{2N} & a_{3N} \\ & & & -1 & 1 \end{bmatrix} \quad C(x) = \begin{bmatrix} 0 & 0 & & & \\ b_{12} & b_{22} & b_{32} & & 0 \\ & & & & \\ & & & & \\ 0 & & b_{1N} & b_{2N} & b_{3N} \\ & & & 0 & 0 \end{bmatrix}$$

with

$$a_{1j} = \frac{\epsilon_{i-1}}{2(\Delta y)^2} ; \quad a_{2j} = -\frac{\epsilon_i + \epsilon_{i-1}}{2(\Delta y)^2} - \frac{u_j}{\Delta x} ; \quad a_{3j} = \frac{\epsilon_i}{2(\Delta y)^2} ,$$

$$b_{1j} = -\frac{\epsilon_{i-1}}{2(\Delta y)^2} ; \quad b_{2j} = \frac{\epsilon_i + \epsilon_{i-1}}{2(\Delta y)^2} - \frac{u_j}{\Delta x} ; \quad b_{3j} = -\frac{\epsilon_i}{2(\Delta y)^2} ,$$

$$i = j ,$$

$$j = 2, \dots, N .$$

The boundary conditions are automatically fulfilled because of the form of the matrix [a] and [b], which gives

$$C_1(x+\Delta x) - C_2(x+\Delta x) = 0 ,$$

$$-C_N(x+\Delta x) + C_{N+1}(x+\Delta x) = 0 .$$

The linear system is solved at each step in order to get $C(x+\Delta x)$. The method of elimination (Ref. 8) is used. The method is programmed in such a way that some of the computations are made only once.

V. INITIAL VALUE

The function $\delta(y-y_1)$ can be developed in a Fourier series (Ref. 5) of period 2.

This series for a period 2ℓ is written

$$\sum_{m=-\infty}^{+\infty} \delta(y-y_1 - 2m\ell) = \frac{1}{2\ell} + \frac{1}{\ell} \sum_{n=1}^{\infty} \cos \left[\frac{n\pi}{\ell} (y-y_1) \right]$$

We want y_1 to have any value between $(0, 1)$ or, y_1 to be in any node j for $j = 2, \dots, N$.

The initial value must also fulfill the boundary conditions and this is done using the symmetries of the problem to exploit the method of images.

This is done in superposing two infinite rows of δ functions of period 2, shifted one with respect to the other, see Fig. 4.

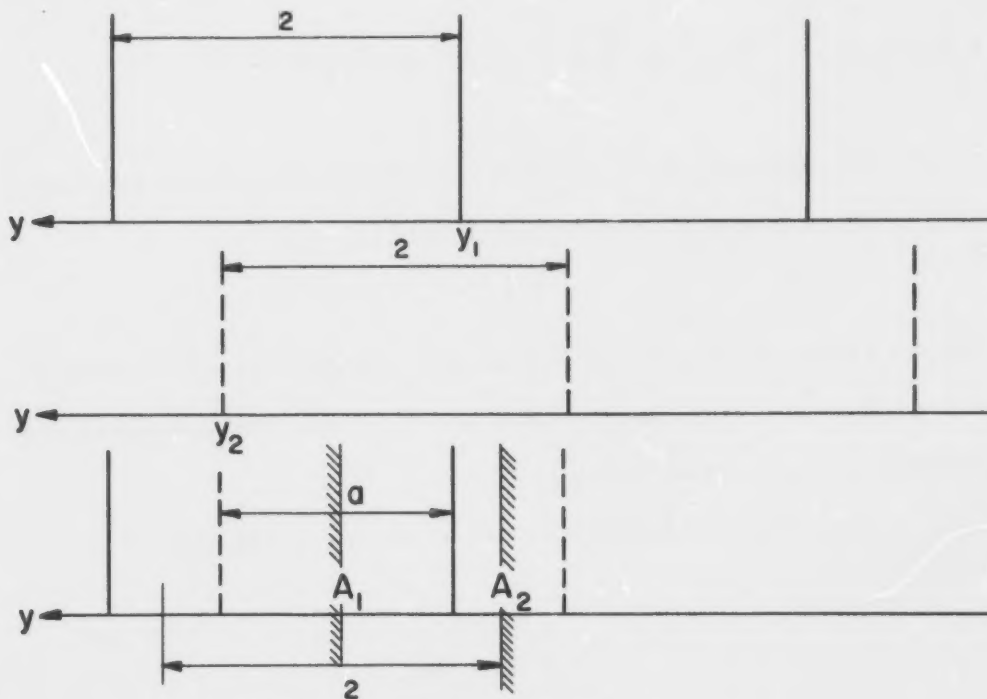


Fig. 4 The geometry of the initial value.

Thus it is possible to vary a , still satisfying the boundary conditions in A_1 and A_2 .

The third function is developed in Fourier series as

$$\begin{aligned} \sum_{m=-\infty}^{+\infty} \delta(y-y_1 - 2m) + \sum_{m=-\infty}^{+\infty} \delta(y-y_2 - 2m) \\ = 1 + \sum_{n=1}^{\infty} \left\{ \cos \left[n\pi (y-y_1) \right] + \cos \left[n\pi (y-y_2) \right] \right\} \end{aligned}$$

Using the discrete set of points j and setting

$$y_1 = y_{j_\delta}$$

we get:

$$\delta_j(j_\delta) = 1 + \sum_{n=1}^P \left\{ \cos \left(n\pi \frac{j_\delta - j}{N-1} \right) + \cos \left(n\pi \frac{3 - j_\delta - j}{N-1} \right) \right\} .$$

P is the frequency for which the Fourier series is truncated.

The numerical method works only if

$$P < N-1 ,$$

and the larger P , the larger the accuracy of the numerical method.

It seems very easy to calculate $\delta(j_\delta)$ but, the numerical computation is much more difficult.

The truncated Fourier series is shown in Fig. 5.

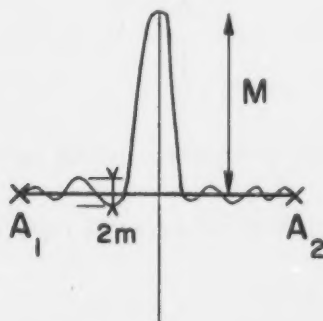


Fig. 5 Series approximation to the δ -function.

We can discuss the results in terms of the ratio $\frac{M}{m}$ versus P . M is the value of the peak and m the amplitude of the first oscillation, (See Table 2). It is easily seen that

$$\left. \begin{array}{l} M = P + 1 \text{ if } P \text{ is even} \\ M = P - 1 \text{ if } P \text{ is odd} \end{array} \right\} \text{ for } \frac{y_1}{d} = 0.5$$

Table 2 Ratios of M/m for $N = 200$.

P	195	196	197	198	199	201
$\frac{M}{m}$	¹ 48.5	¹ 65.3	¹ 98.5 ₂ 99	¹ 10 ⁶ ₂ 10 ¹⁵	¹ 10 ⁶ ₂ 10 ¹⁵	² 100

¹ without optimisation of the computations

² with optimisation of the computations

The computations are made in double precision.

The difference between the method 1 and the method 2 is an optimisation of the computations. This optimisation is made first by diminishing the number of elementary operations in each step, second by improving the order in which those operations are made.

The table shows a strange behavior for $P = N-2$ and $P = N-1$. The explanation was extremely difficult to find.

The comparison of the methods 1 and 2 shows that the accuracy of the computations has a strong influence on the result.

The error involved in each computation of the cosine function generates a phase lag between the harmonics. Therefore for some values of P the higher order harmonics happen to be in phase with the nodes j on which the values of the series are read.

For $P=198$ and $P=199$, the values on each side of a symmetry point like A_1 are $+10^{-13}$ and -10^{-13} . (See Fig. 6.)

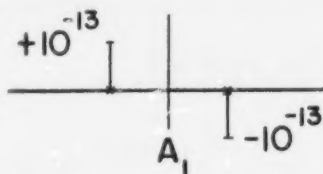


Fig. 6 Side lobe magnitude for $P = 198; 199$.

This hypothesis has been checked, using $P=199$ and shifting the nodes j by $\frac{\Delta y}{2}$. The value of $\frac{M}{m}$ read is then 100.

If the δ -function is set in the middle ($y_1 = \frac{d}{2}$), then the boundary conditions are fulfilled because $\pm 10^{-13}$ is almost zero and the numerical computation gives the same results for the concentration for $P = 197$ and $P = 198$ (see Programs JØB 762 and JØB 673). This give a second proof that a phase lag due to precision problems is the only reason for the behavior shown in Table 2.

If the δ -function is set close to the bed or close to the free surface, the boundary conditions have to be fulfilled with much more accuracy. Therefore $P = 198$ gives rise to trouble in the region where we want the solution of the numerical method to behave well.

The value $P = 197$ is therefore kept on for the computations, together with $N = 200$.

VI. VELOCITY PROFILE

The logarithmic velocity profile is used. There is no problem for $y = 0$ because the value of the velocity is needed only at the nodes j and there is no node j on the bottom.

For $N = 200$, the node $j=N$ is at $y/d \sim 2.5 \times 10^{-3}$ and the value of the logarithm is reasonable.

The velocity profile for turbulent flow is

$$U = \bar{U} + \frac{U_*}{K} \ln \frac{y}{d} + \frac{U_*}{K}, \quad (4)$$

$\bar{U}$ = mean velocity,

U_* = shear velocity on the bed ,

K = von Karman constant .

If the friction factor f is introduced as

$$\frac{\bar{U}}{U_*} = \sqrt{\frac{8}{f}}.$$

then U is written in the dimensionless form

$$\frac{U}{U_*} = \sqrt{\frac{8}{f}} + \frac{1}{K} (\ln \frac{y}{d} + 1).$$

The two parameters K and f are chosen arbitrarily. On the grid, Eq. (4) is written

$$\frac{U_j}{U_*} = \sqrt{\frac{8}{f}} + \frac{1}{K} \left[\ln \left(\frac{N+2 - 2j}{2(n-1)} + \frac{1}{2} \right) + 1 \right].$$

VII. DIFFUSIVITY

The hypothesis that the diffusivity of the substance ϵ_y is proportional to the diffusivity of momentum ϵ_m is made; i. e. $\epsilon_y = \alpha \epsilon_m$.

Using the logarithmic profile for the velocity, the diffusivity can be analytically computed.

The result, which is classical (Ref. 10) is

$$\frac{\epsilon_y}{U_* d} = \alpha K \left(1 - \frac{y}{d} \right) \frac{y}{d}$$

This parabolic profile leads to a contradiction at the bottom where it is zero. That means that ϵ_m is also zero at the bed and therefore τ_o is also zero, which is impossible. However, no better assumption can be made and there is no way to get a value for ϵ_y on the bed. This assumption is not very troublesome as long as the injected substance does not react with the bed material.

Instead of using α as a parameter for ϵ_y , the average value $\overline{\epsilon_y}$ over the depth is introduced. We get, in a dimensionless form

$$\frac{\epsilon_y}{U_* d} = 6 \frac{\overline{\epsilon_y}}{U_* d} \left(1 - \frac{y}{d}\right) \frac{y}{d}.$$

On the grid i , this is written

$$\frac{\epsilon_i}{U_* d} = 6 \frac{\overline{\epsilon_y}}{U_* d} \left[\frac{1}{4} - \left(\frac{N+1 - 2i}{2(N-1)} \right)^2 \right]$$

Remark 1:

Only the values of $\frac{U}{U_*}$ and $\frac{\epsilon}{U_* d}$ at the nodes j and i need be known to apply the numerical method; therefore any experimental profiles could be used instead of the analytical expressions.

Remark 2:

The complete equation could be solved in the same way as it is also parabolic.

The difference scheme is

$$[a'] C(x + \Delta x) = [b'] C(x) + [c'] C(x - \Delta x).$$

The first integration from $x = 0$ to $x = \Delta x$ can be made using the simplified equation $\left[\text{without } \frac{\partial}{\partial x} \left(\epsilon_x \frac{\partial C}{\partial x} \right) \right]$ as we did. Then the three level difference scheme could be used.

VIII. PROGRAM CHECK

It is possible to compare the results with an analytical solution for the case where U and ϵ_y are constant.

At the beginning, the influence of the free surface and the bed can be neglected.

Let us call $C_{\max}$, the maximum value of C in a given section. The analytical solution is (Ref. 7) a Gaussian curve and the classical result is

$$C_{\max} \propto \frac{1}{\sqrt{x}}.$$

Let us call C_0 , the value of C such that

$$\frac{C_0}{C_{\max}} = e^{-\frac{1}{2}},$$

in a given section.

Then the classical result is

$$\sigma^2 \propto x,$$

where σ is the distance from the ordinate of $C_{\max}$ to the ordinate of C_0 .

To check the numerical analysis against these analytical results, data in Table 3 were read from program JØB 113; the results show very good confirmation of the numerical analysis. However, the best proof of the accuracy of the program is that

$$\frac{C}{S/\bar{U}d} = 1 \text{ at } \frac{x}{d} = 191$$

The amount of substance injected per unit time is preserved even far from the source (see program JØ B 113).

Table 3. Result of JØB 113. (u , ϵ_y are constant and $y_1 = 0.5$)

$\frac{x}{d}$	.1	.2	.3	.6	1.	6.
$\frac{C_{\max}}{S/\bar{U}d}$	18.51	13.06	10.65	7.52	5.83	2.38
$\frac{\sigma}{\Delta y}$	4.1	6.2	7.4	10.5	13.5	33.3
$\frac{C_{\max} \sqrt{x}}{S/\bar{U}d^{3/2}}$	5.85	5.85	5.84	5.83	5.83	5.83
$\frac{\sigma^2 d}{(\Delta y)^2 x}$	168	185	183	183.5	182	185

IX. RESULTS FOR DIFFERENT CASES

The program is run for different values of y_1 shown in Table 4, using variable coefficients based on the following data:

$$f = \text{friction factor} = 9 \times 10^{-3},$$

$$K = \text{von Karman constant} = 0.40,$$

$$\frac{\bar{\epsilon}}{U_* d} = \text{dimensionless mean diffusivity} = 0.07.$$

We can check first that the boundary conditions are always fulfilled at A_1 and A_2 even for the case $j_\delta = 199$. This allows us to use the program to solve the problem of a source practically on the bed. The results are shown on the plots identified by the same JØB number in a self-explanatory way.

Table 4. Variation of source height.

Program Identification	JØB 37	JØB 112	JØB 762	JØB 111	JØB 563	JØB 564
j_δ = j-value at source	197	70	101	150	200	2
$\frac{y_1}{d}$ = relative source position (from bed)	.0175	.6558	.5000	.2538	.0025	.9975

X. CONCLUSION:

It is not our purpose to discuss the physical phenomena at length but some conclusions can be given:

(1). The variation of the diffusivity has a strong influence on the behavior of the solution. The peak concentration in a given profile tends to move to the region where the diffusivity is smaller.

The following explanation can be given (Fig. 7). The higher the diffusivity, the more substance is carried away. Thus, when the diffusivity is higher (as in lower part of Fig. 7) more substance is removed from that side of the profile than from the other. This result is reasonable and is shown on the plot JØB 112. The substance is injected in a domain where the velocity profile is rather flat.

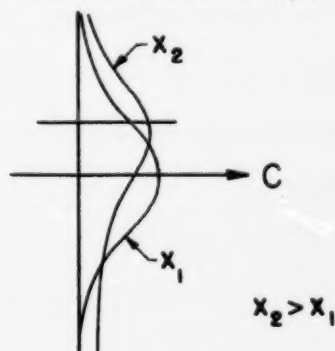


Fig. 7. Concentration profiles.

(2) The peak concentration moves to a region where the velocity is lower. This result has already been found in experiments (Ref. 12).

In the JØB 111, both the diffusivity gradient and the velocity gradient tend to move the peak concentration down to the bed.

In the JØB 112, the velocity gradient (small) works against the diffusivity gradient which tends to move the peak concentration to the surface.

This addition or subtraction of the two effects is easily seen on the respective plots.

The peak concentration reaches the bed for $x/d \sim 20$ in the first case and reaches the surface for $x/d \sim 40$ in the second one.

The interesting behavior shown on the plot JØB 112 should be checked by an experiment. It would be a good way to test if the diffusivity is actually a parabola.

(3) In the case of an injection at the bed, the strong gradient of velocity and the gradient of diffusivity prevent the substance from spreading at the beginning. The same behavior occurs in the numerical computations and the result is non-conservation of tracer. (See JOB 563 $\frac{C}{S/\bar{U}d} = .58$ for $x/d = 191$.) Therefore j_δ should not be larger than $N-5$ (but N can be increased to be closer to the bed, see JØB 37).

This program could be used to make a complete investigation of the problem and to design experiments. It could be a very useful tool for further studies.

Acknowledgment

This paper was done as a student assignment by Jean F. Coudert for the course, Env 112b, Hydrologic Transport Processes, given by Professor Norman H. Brooks in the third term 1969-70.

It has been reproduced with funds from Federal Water Quality Administration Grant No. 16070 DGY in order to make the work available to others.

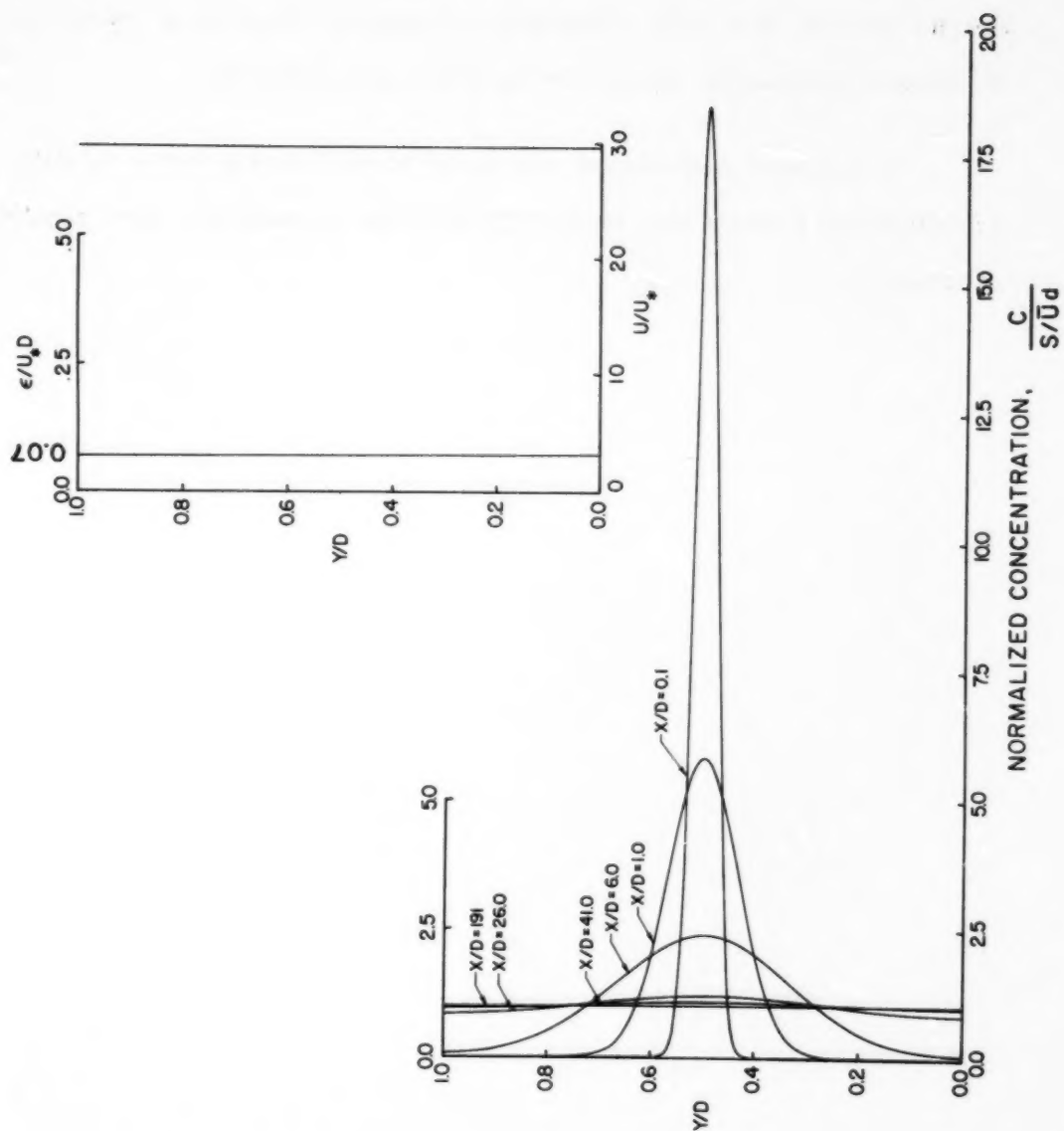


Figure 8. Results of Job 113. $U = \text{const.}$; $\epsilon = \text{const.}$ $y_1/d = .5$.

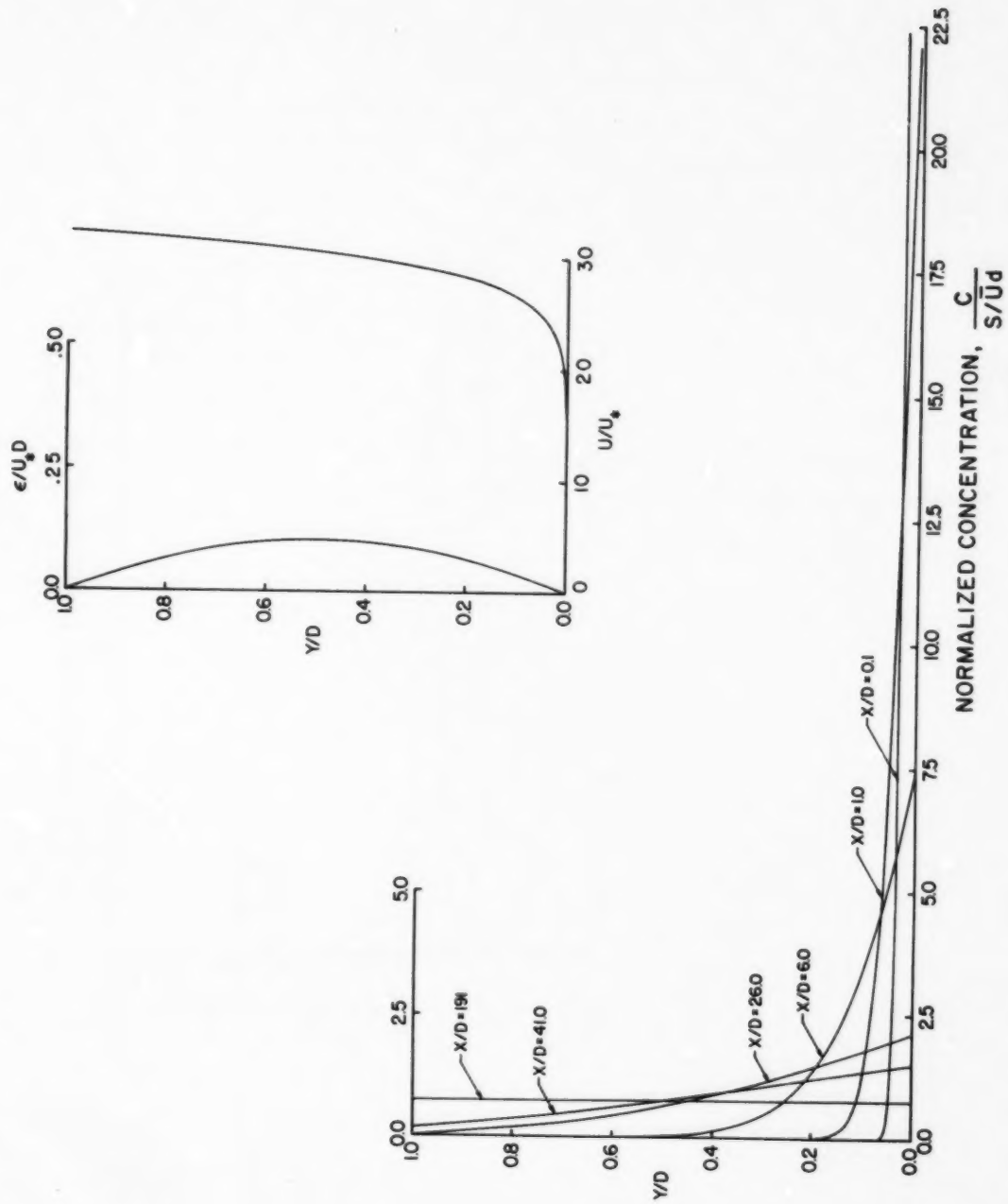


Figure 9a. JØB 37.

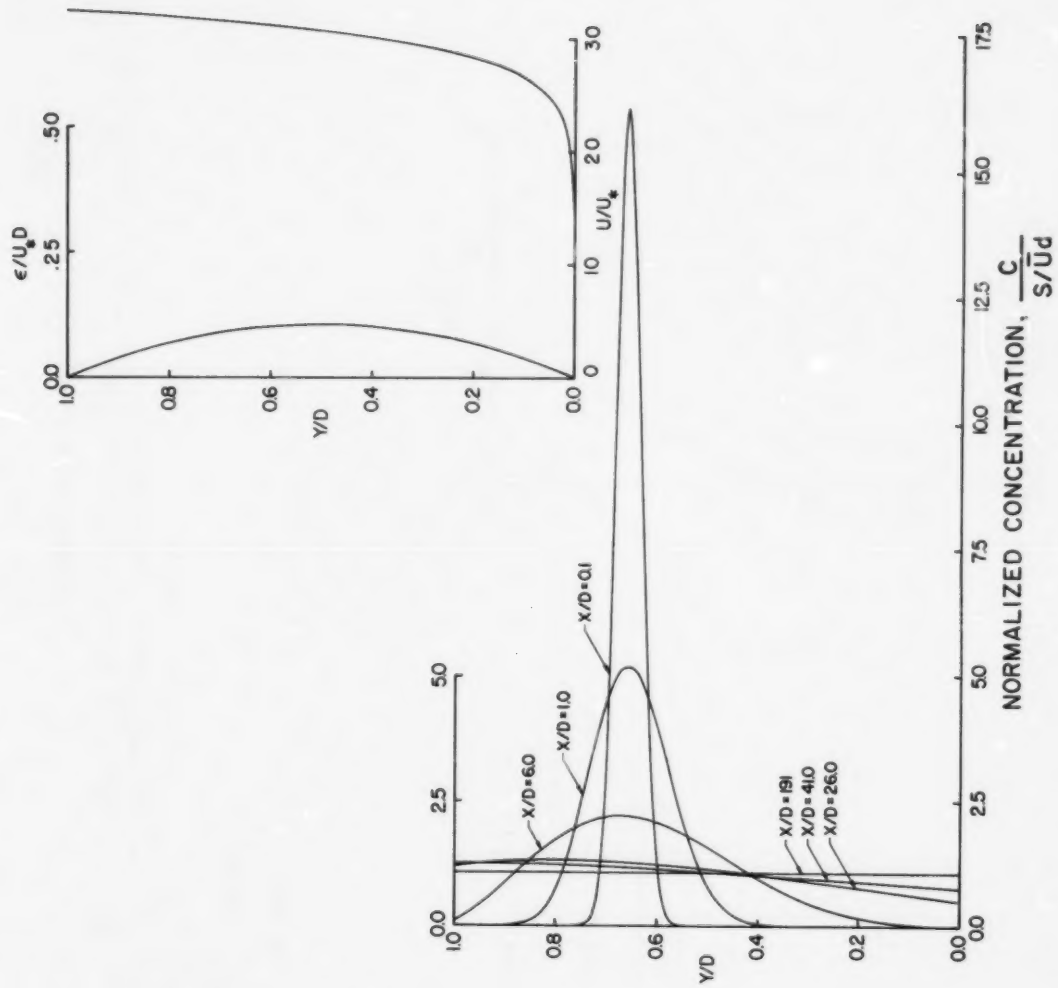


Figure 9b. JØB 112.

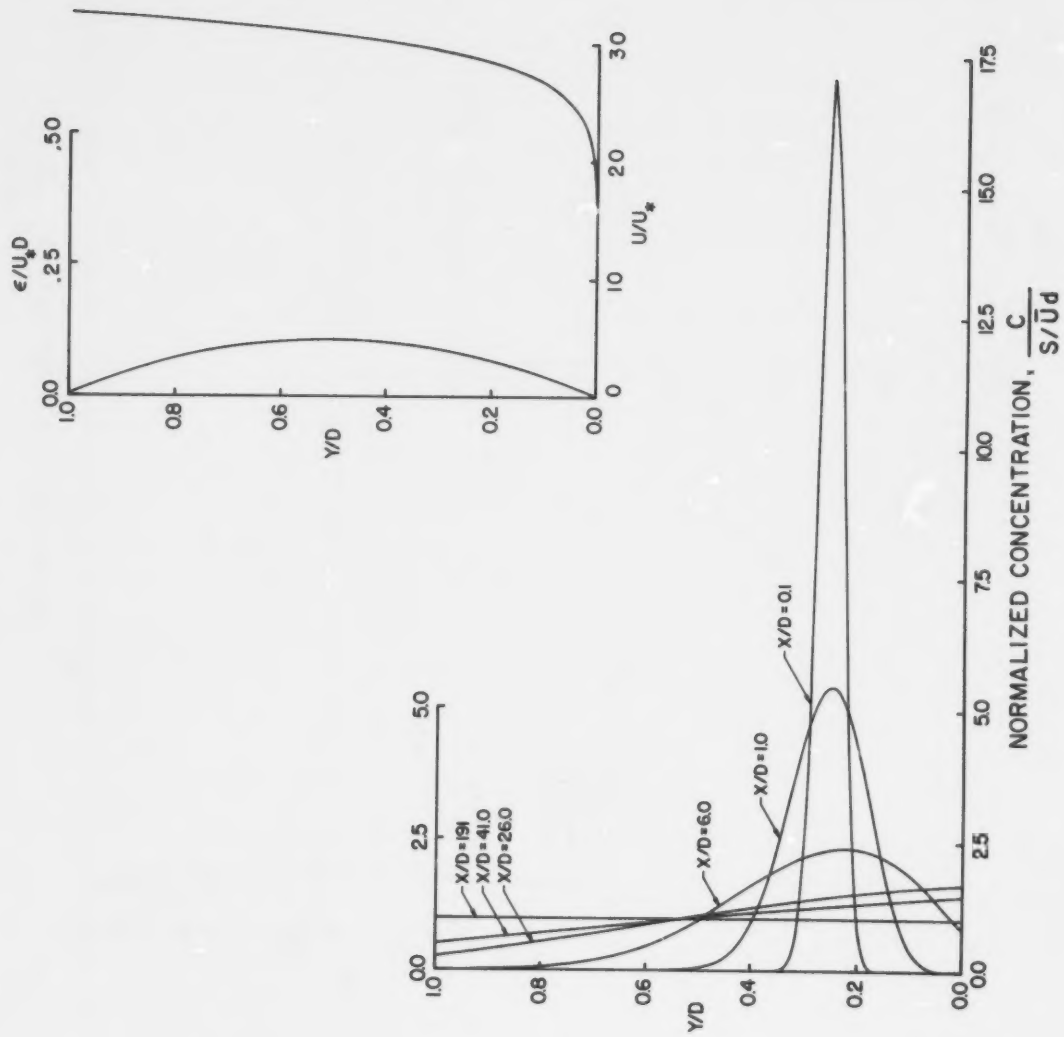


Figure 9c. JØB 762.

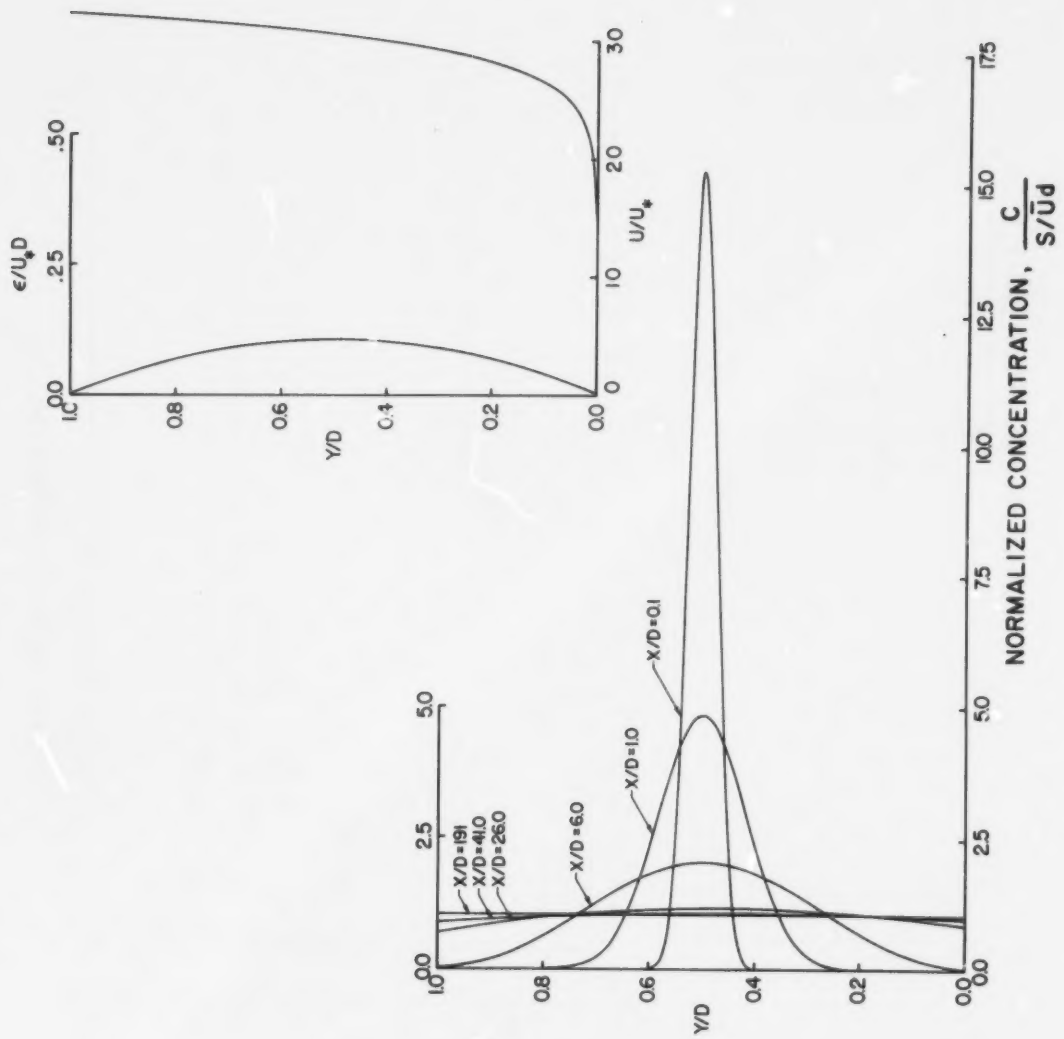


Figure 9d. JØB 111.

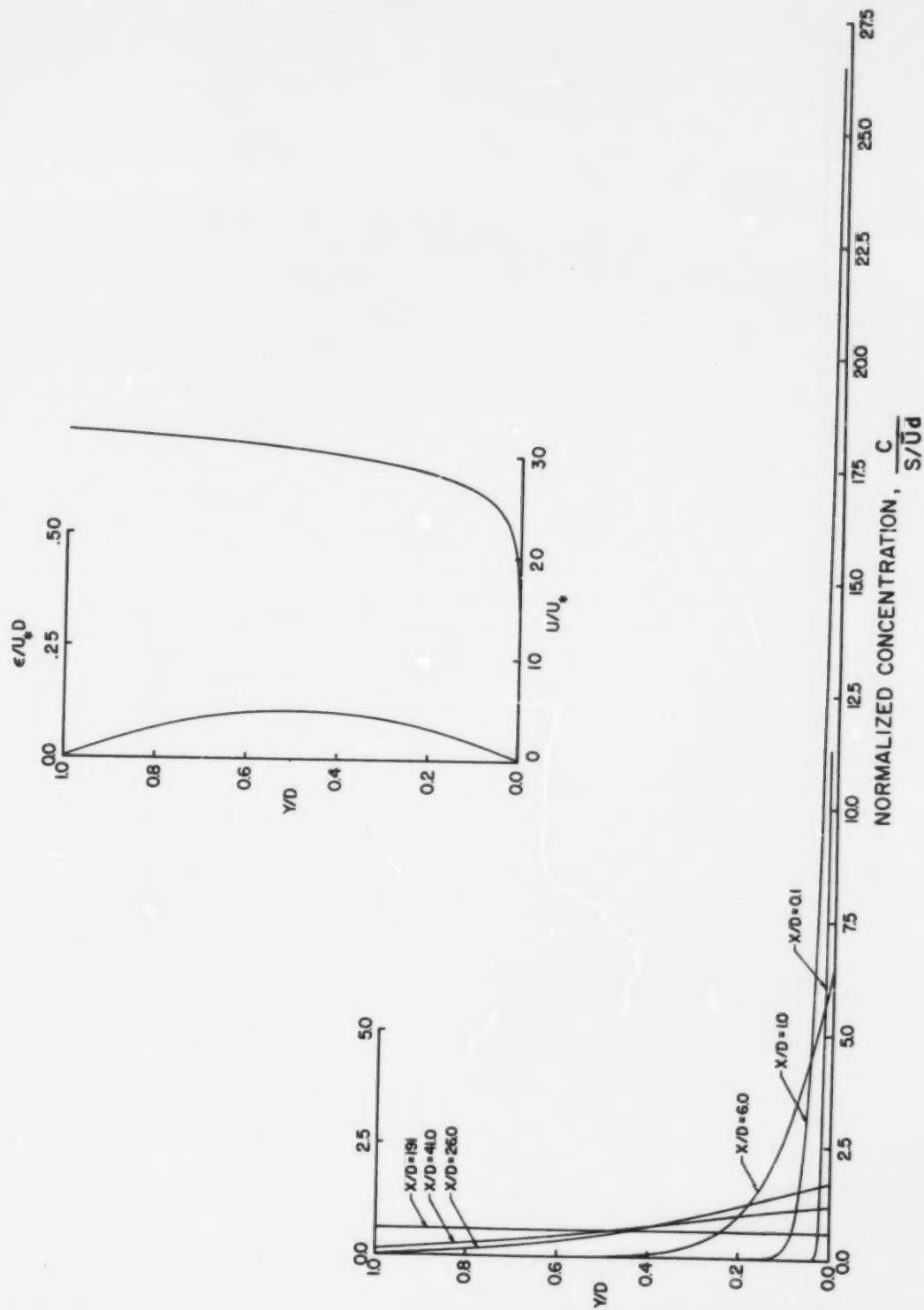


Figure 9e. JØB 563.

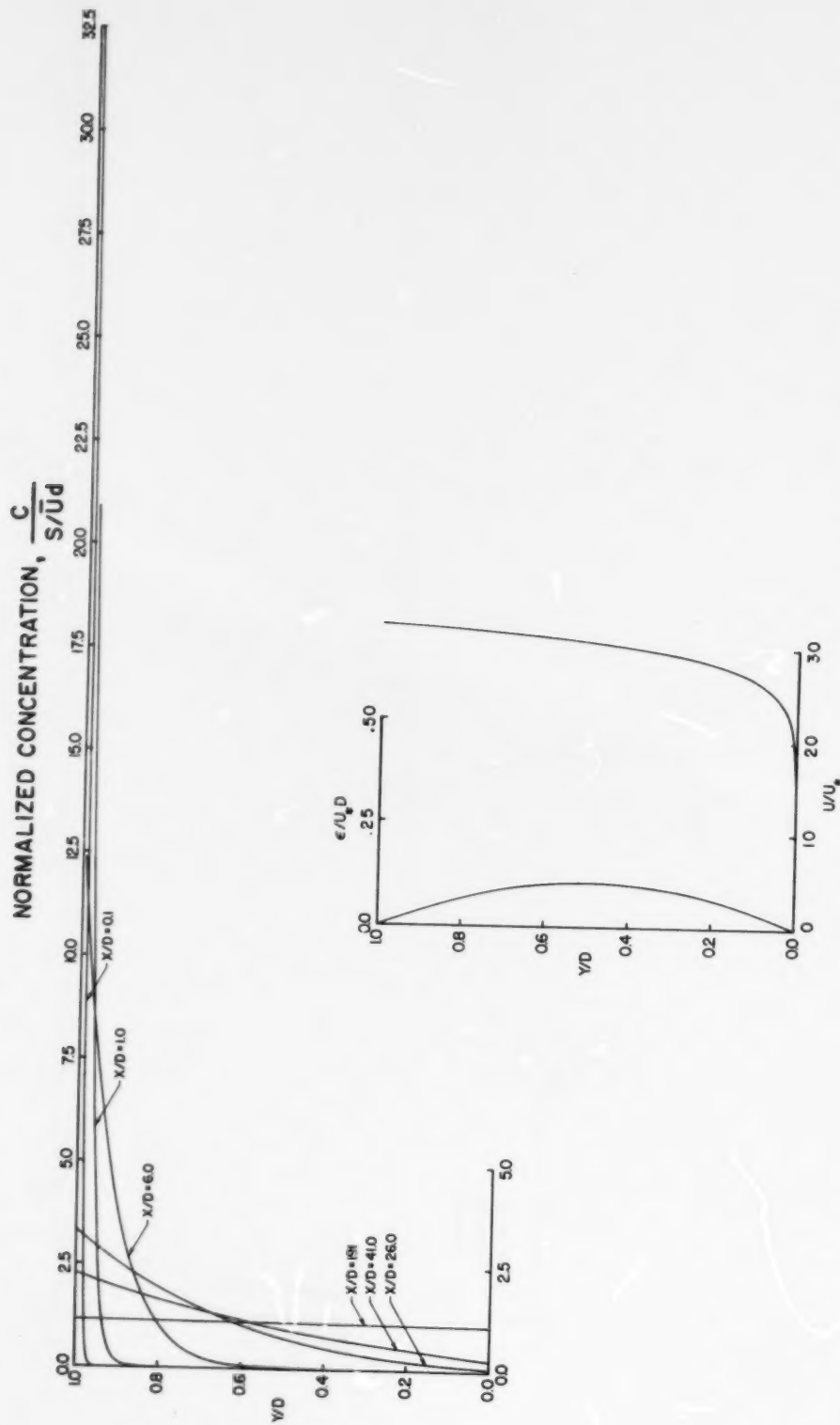


Figure 9f. JØB 564.

REFERENCES

1. Saffman, P. G. (1962), "The Effect of Wind Shear on Horizontal Spread from an Instantaneous Ground Source", Quart. J. Roy. Meteorol. Soc., pp. 382-393.
2. Sutton, O. G. (1953), Micrometeorology, McGraw-Hill Book Company, New York, pp. 280-283.
3. Pasquill, F. (1962), Atmospheric Diffusion, D. Van Nostrand Co., Ch. 4.
4. Kreiss, H. O. (1970), "Numerical Solutions of Partial Differential Equations," Lecture Notes, AMa 204 - Spring Term, California Institute of Technology.
5. Lighthill, M. J. (1958), Introduction to Fourier Analysis and Generalised Functions, Cambridge Mono. on Mech. and App. Math. University Press, pp. 1-7, pp. 58-75.
6. Okubo, A. and Karweit, M. J. (1969), "Diffusion from a Continuous Source in a Uniform Shear Flow", Limnology and Oceanography, Vol. 14, No. 4, pp. 514-520.
7. Carslaw, H. S. and Jaeger, J. C. (1960), Conduction of Heat in Solids, Oxford at the Clarendon Press.
8. Franklin, J. N. (1968), Matrix Theory, Prentice Hall, Inc., pp. 195-202.

9. Taylor, G. I. (1954), "The Dispersion of Matter in Turbulent Flow Through a Pipe", Proc. Roy. Society of London, 223, pp. 446-468.
10. Fisher, H. B. (1966), "Longitudinal Dispersion in Laboratory and Natural Stream", Report KH-R-12, W. M. Keck Laboratory of Hydraulics and Water Resources, California Institute of Technology, pp. 5-19.
11. Richtmyer, R. D. and Morton, K. W. (1967), Difference Methods for Initial Value Problems, Interscience Publishers, pp. 185-195.
12. Okoye, J. K., (1970), "Characteristics of Transverse Mixing in Open-Channel Flows", Report KH-R-23, W. M. Keck Laboratory of Hydraulics and Water Resources, California Institute of Technology,

APPENDIX A

PROGRAM (IN FORTRAN IV)

A-i

```

C      N MUST BE EVEN
      IMPLICIT REAL*8(O)
      REAL K
      INTEGER SWITCH,STEP,P,STEP1,STEP2,R,STEP3
      DATA DPI/3.141592653589793/
      DIMENSION C(401),A(401,3),B(401,3),X(401),Z(401),ZZ(401),CC(401),
1 Y(401),ZZY(401),DD(3),YMAX(401),X(401)
      COMMON EPSR,N,K,F
      COMMON/COMPLC/ITEST,XLENGTH,YLENGTH
C      DATA SPECIFICATION
      NSTOP=0
      N=200
      K = 0.375
      NWRT=1
      EPSR = 0.0625
      DX1 = 0.01
      DX2 = 0.1
      DX3 = 0.2
      DX4 = 0.3
      DX5 = 0.2
      STEP1 = 20
C      STEP2 = 55
      STEP2 = 36
      STEP3 = 48
      STEP4 = 64
      STEP5 = 69
      MAX1 = 100
C      MAX2 = 165
      MAX2 = 72
      MAX3 = 96
      MAX4 = 123
      MAX5 = 345
      OK = 1
      IRUN = 509
1001 READ (5,101) F, JOELTA
101  FORMAT (F10.0,I5)
      WRITE (6,103) IRUN
103  FORMAT ('1  RUN NO ',I3//)
C      END DATA SPECIFICATION
      CD(1)=0
      CD(2)=0
      CD(3)=1
      XLENGTH=45
      YLENGTH=10
      ITEST=1
      K=1
      X(1)=0
      P=N-3
      MAX=MAX1
      SWITCH=1
      XO=0
      DX=DX1
      STEP=STEP1
      ITER=1
      UDY=1.0000/DFLOAT(N-1)
      DY=UDY

```

```

YMAX(1)=1-(JDELTA-1.5)*DY
NN=N+1
DO 30 I=1,N
IF(I.EQ.1)GO TO 31
Y(I)=1-(I-1.5)*DY
Z(I)=U(I)
31 ZZ(I)=EPS(I)
ZZY(I)=1-(I-1)*DY
30 CONTINUE
WRITE(6,502)
502 FORMAT('  VALUES OF VELOCITY U/U* FROM NODE 2 TO NODE N'///)
WRITE(6,501)(Z(I),I=2,N)
WRITE(6,503)
503 FORMAT('1 VALUES OF DIFFUSIVITY EPS/U*.D FROM NODE 1 TO NODE N'///)
11
WRITE(6,501)(ZZ(I),I=1,N)
N1=N-1
CALL LABEL(0.0,0.0,0.0,1.0,10.,5,'Y/D',3,1)
CALL LABEL(0.0,0.0,0.0,30.,5.75,3,'U/U*',4,0)
CALL LABEL(0.0,10.,0.0,.5,5.,2,'EPS/U*.D',-8,0)
CALL XYPLUT(N1,Z(2),Y(2),0.0,200.,0.0,1.0,00,0)
CALL XYPLUT(N,ZZ(1),ZZY(1),0.0,4.5,0.0,1.0,00,0)
C
INITIAL VALUE
WRITE(6,504)
504 FORMAT('1 INITIAL VALUES OF THE CONCENTRATION C/(S/UMEAN.D)'///)
NNN=2*N
DO 2 J=1,NNN
CC(J)=1
CJ1=CPI*ODY*DFLOAT(JDELTA-J)
CJ2=CPI*ODY*DFLOAT(3-JDELTA-J)
DO 1 I=1,P
OI=DFLOAT(I)
OC(J)=OC(J)+OCOS(OI*CJ1)+OCOS(OI*CJ2)
1 CONTINUE
2 CONTINUE
DO 35 J=1,NNN
35 C(J)=CC(J)
WRITE(6,501)(C(I),I=1,NNN)
501 FORMAT(' '////'(' ',1PE15.6,3X,1PE15.6,3X,1PE15.6,3X,1PE15.6,3X,1PE
115.6,3X,1PE15.6/))
IF(NSTOP.EQ.1) STOP
WRITE(6,500)DX,STEP
C
MATRIX COMPUTATION
A(1,1)=0
A(1,2)=1
A(1,3)=-1
A(NN,1)=-1
A(NN,2)=1
A(NN,3)=0
11 DO 3 J=2,N
A(J,1)=EPS(J-1)/(2*DY*DY)
A(J,2)=-((EPS(J)+EPS(J-1))/(2*DY*DY)+U(J)/DX)
A(J,3)=EPS(J)/(2*DY*DY)
B(J,1)=-EPS(J-1)/(2*DY*DY)
B(J,2)=(EPS(J)+EPS(J-1))/(2*DY*DY)-U(J)/DX
B(J,3)=-EPS(J)/(2*DY*DY)

```

```

      2 CONTINUE
C     BEGIN INTEGRATION
      M=1
      DO 10 I=1,MAX
        INDEX=STEP*M
        DO 4 J=2,N
          4 W(J)=B(J,1)*C(J-1)+B(J,2)*C(J)+B(J,3)*C(J+1)
          W(1)=0
          W(NN)=0
          CALL EQSOLV(NN,A,W,C,SWTCH)
          SWTCH=0
          IF(1.NE.INDEX) GO TO 10
          M=M+1
          R=R+1
          CMAX=0
          DO 15 II=1,NN
            15 CMAX=AMAX1(CMAX,C(II))
            LMAX=0
            DO 16 II=1,NN
              LMAX=LMAX+1
              IF(C(II).EQ.CMAX) GO TO 17
            16 CONTINUE
            17 YMAX(F)=1-(LMAX-1.5)*DY
            X(R)=I*DX+X0
            WRITE(6,505)X(R),CMAX,YMAX(R)
505   FORMAT(' '///' X/D=',1PE15.6,' CMAX=',1PE15.6,' YMAX/D=',
            11PE15.6///)
            WRITE(6,501)(C(J),J=1,NN,NWRT)
C     XPLT = 0.2
      XPLT = 3.20
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        6CALL XYPLT(N1,C(2),Y(2),-9.,36.,0.0,1.0,DD,0)
      IF (JK.GE.1) GO TO 10
      XPLT=1
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        1CALL XYPLT(N1,C(2),Y(2),-17.,28.,0.0,1.0,DD,0)
      XPLT = 17.5
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        2CALL XYPLT(N1,C(2),Y(2),-23.,22.,0.0,1.0,DD,0)
      XPLT = 36.7
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        3CALL XYPLT(N1,C(2),Y(2),-29.,16.,0.0,1.0,DD,0)
      XPLT = 55.9
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        4CALL XYPLT(N1,C(2),Y(2),-35.,10.,0.0,1.0,DD,0)
      XPLT = 75.1
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        5CALL XYPLT(N1,C(2),Y(2),-39.,6.,0.0,1.0,DD,0)
      XPLT = 102.7
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        6CALL XYPLT(N1,C(2),Y(2),-43.,2.,0.0,1.0,DD,0)
      XPLT = 144.1
      IF(ABS(X(R)-XPLT).LE.1.0E-2)
        7CALL XYPLT(N1,C(2),Y(2),-47.,1.,0.0,1.0,DD,0)
      10 CONTINUE
C     IF(ITER.GE.2) GO TO 1000

```

```

IF(ITER.GE.2) GO TO 2000
X0=X(R)
ITER=2
MAX=MAX2
STEP=STEP2
DX=DX2
SWTCH=1
WRITE(6,500)DX,STEP
500 FORMAT(1H1,'COMPUTATIONS WITH DX=',1PE15.6////' VALUES OF THE CONCE
INTRATION EACH ',I3,'*DX DISTANCE'////)
GO TO 11
1000 IF(ITER.GE.3) GO TO 1500
X0=X(R)
ITER=3
MAX=MAX3
STEP=STEP3
DX=DX3
SWTCH=1
WRITE(6,500)DX,STEP
GO TO 11
1500 IF(ITER.GE.4) GO TO 1600
X0=X(R)
ITER = 4
MAX = MAX4
STEP = STEP4
DX = DX4
SWTCH = 1
WRITE(6,500) DX,STEP
GO TO 11
1600 IF(ITER.EQ.5) GO TO 2000
X0=X(R)
ITER = 5
MAX = MAX5
STEP = STEP5
DX = DX5
SWTCH = 1
WRITE(6,500) DX,STEP
GO TO 11
C2000 CALL LABEL(9.,0.0,0.0,5.,5.,2,'X/D=.200',8,0)
2000 CALL LABEL(9.,0.0,0.0,5.,5.,2,'X/D=.8.20',8,0)
IF (OK.GE.1) GO TO 200
CALL LABEL(17.,0.0,0.0,5.,5.,2,'X/D=1.00',8,0)
CALL LABEL(23.,0.0,0.0,5.,5.,2,'X/D=17.5',8,0)
CALL LABEL(29.,0.0,0.0,2.,2.,2,'X/D=36.7',8,0)
CALL LABEL(35.,0.0,0.0,2.,2.,2,'X/D=55.9',8,0)
CALL LABEL(39.,0.0,0.0,2.,2.,2,'X/D=75.1',8,0)
CALL LABEL(43.,0.0,0.0,2.,2.,2,'X/D=102.7',9,0)
CALL LABEL(47.,0.0,0.0,2.,2.,2,'X/D=144.1',9,0)
200 CALL LABEL(9.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
IF (OK.GE.1) GO TO 201
CALL LABEL(17.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL LABEL(23.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL LABEL(29.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL LABEL(35.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL LABEL(39.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL LABEL(43.,0.0,0.0,1.0,10.,5,'Y/D',0,1)

```

```
CALL LABEL(47.,0.0,0.0,1.0,10.,5,'Y/D',0,1)
CALL SYSEND(-1,0)
XLNGTH=15
ITEST=1
CALL LABEL(0.0,0.0,0.0,1.0,10.,5,'Y/D',3,1)
CALL LABEL(0.0,0.0,0.0,144.1,15.,10,'X/D',3,0)
CALL XYPLUT(R,X(1),YMAX(1),0.0,191.,0.0,1.0,00,1)
201 CK = CK + 1
    IRUN = IRUN + 1
    IF (CK.GT.4) GO TO 102
    GO TO 1001
102 STOP
END
```

```
C      FUNCTION EPS(J)
      EPS IS THE DIFFUSION COEFFICIENT
      COMMON EPSR,N,K,F
      EPS=6*EPSR*(.25-((N+1.-2.*J)/(2.*(N-1.)))**2)
      RETURN
      END
```

C FUNCTION U(J)
 U IS THE LOCAL VELOCITY
 REAL N
 COMMON EPSR,N,K,F
 U=SQRT(8/F)+(1 + ALUG((N+2.-2.*J)/(2.*(N-1.))+.5))/K
 RETURN
 END

```

SUBROUTINE EQSOLV(Q,A,B,Y,INIT)
INTEGER Q
DIMENSION A(401,3),B(1),Y(1)
DIMENSION U(401,3),D(401,3)
IF(INIT.EQ.0) GO TO 30
U(1,2)=A(1,2)
U(1,3)=A(1,3)
DO 20 I = 2, Q
  I1 = I - 1
  D(I,2) = 1.
  U(I,3) = A(I,3)
  D(I,1) = A(I,1) / U(I1,2)
  U(I,2) = A(I,2) - D(I,1)*U(I1,3)
  IF (ABS(U(I,2)).LT.1.E-09) GO TO 35
20 CONTINUE
30 Y(1) = B(1)
DO 10 I=2,Q
  I1 = I - 1
10 Y(I) = B(I) - D(I,1) * Y(I1)
  Y(Q) = Y(Q) / U(Q,2)
  I=Q
25 I1 = I - 1
  Y(I1) = (Y(I1) - U(I1,3) * Y(I)) / U(I1,2)
  I = I - 1
  IF(I .EQ. 1) GO TO 15
GO TO 25
15 RETURN
35 WRITE(6,7)
7 FORMAT(//5X'A IS A MATRIX WHICH REQUIRE PIVOTING OR IS SINGULAR')
STOP
END

```

APPENDIX B

SAMPLE PROBLEM

The following variables must be given:

N: number of points (must be even) = 200

P: truncation number

P must be chosen as large as possible with respect to the condition of $P < N-1$

P must be chosen carefully so that the values of $\frac{C}{S/\bar{U}_d}$ at nodes 1, 2 at the free surface and N, N + 1 at the bed are respectively equal. $\frac{C}{S/\bar{U}_d}$ for the initial value is printed out.

P = 197

JDELTA: No. of the node where the δ function is set

$$2 \leq JDELTA \leq N-5$$

DX1, DX2, DX3: first, second and third increment used.

$$DX1 < DX2 < DX3$$

$$\frac{DX3}{DY} \cdot \frac{\bar{\epsilon}}{\bar{U}_d} < .5 \text{ where } DY = \frac{d}{N-1}; d=1$$

If C is the exact solution of the diffusion equation and $\tilde{C}$ the solution given by the numerical method, then

$$|C - \tilde{C}| \leq K (DX)^2$$

$$\text{where } K \propto \frac{1}{\sqrt{X}}$$

Thus, if accuracy is needed very close to the source DX1 has to be taken very small

$$DX1 = .01; DX2 = .1; DX3 = .5$$

APPENDIX B (Continued)

STEP 1, STEP 2, STEP 3, For the computations using DX1, DX2, DX3,
 results are printed every STEP 1 * DX1, STEP 2 * DX2,
 STEP 3 * DX3 times; STEP 1 = 10; STEP 2 = STEP 3 = 50.

MAX1, MAX2, MAX3 total number of iterations using DX1, DX2, DX3

MAX1 = 100, MAX2 = 400, MAX3 = 300

For the determination of P from the printed initial values of

$\frac{C}{S/\bar{U}d}$, NSTOP can be set to 1 in order to stop the program after this
 printing.

K = von Karman constant = .4

F = bed friction factor = 9×10^{-3}

EPSR = $\frac{\bar{\epsilon}}{U_* d}$ = dimensionless diffusivity = .07

NWRT = Concentrations are printed from j = 1 up to j = N+1 step

NWRT = 1 all the concentrations are printed.